

Plataforma de telemonitoreo IoT basada en algoritmos de detección de patologías cardíacas*

María Paula Bonomini^{1,2}, Cristian Mateos^{1,3}, Matías Hirsch^{1,3}, Lorenza Vecino Schandy⁴, Pedro Barrera⁴, Lucas Grana⁵, and Sergio Liberczuk^{6,7}

¹ Consejo Nacional de Investigaciones Científicas y Técnicas

² Instituto Argentino de Matemática "Alberto P. Calderón" (IAM)

³ Instituto Superior de Ingeniería de Software Tandil (ISISTAN)

⁴ Instituto Tecnológico de Buenos Aires (ITBA)

⁵ VirtualSense S.A.

⁶ Centro de Altos Estudios en Tecnología Informática - Facultad de Tecnología Informática - Universidad Abierta Interamericana

⁷ Instituto de Ingeniería y Agronomía - Universidad Nacional Arturo Jauretche
Corresponding author: `cristian.mateos@isistan.unicen.edu.ar`

Abstract. El telemonitoreo permite obtener información de rutina de la condición del paciente con el fin de seguimiento remoto. Las plataformas de monitoreo implementan via sistemas de software mecanismos que monitorean a distancia alteraciones en los signos vitales, lo que permite detectar las descompensaciones en etapas incipientes y facilitar su tratamiento. Desde VirtualSense S.A. y con investigadores CONICET del IAM e ISISTAN se está desarrollando la primer y única plataforma de telemonitoreo integral que combina IoT e IA del país. La plataforma permite que pacientes con diferentes patologías crónicas -actualmente insuficiencia cardíaca- reciban una atención adecuada desde el hogar. El proyecto comprende tres objetivos técnicos: a) desarrollar una aplicación móvil que controla dispositivos de medición, incluido un ECG, b) producir algoritmos de procesamiento de señales biomédicas y en particular detección de fibrilación auricular, y c) generar un sistema de recomendación que automatiza la dosificación de medicamentos basado en reglas aportadas por expertos en cardiología. En este artículo, se reportan resultados preliminares del proyecto, con foco en describir el diseño y la evaluación de los algoritmos de detección de insuficiencias cardíacas, que han alcanzado alentadores resultados en términos de las Exactitud, Precisión, Especificidad y Sensibilidad.

Keywords: Telemedicina · Telemonitoreo · ECG · Aprendizaje de Máquina · Plataforma de telemonitoreo

1 Introducción

De acuerdo a las definiciones del Ministerio de Salud de la República Argentina, la telesalud involucra "la entrega de servicios de salud usando las TICs, es-

* Financiado por Ministerio de Ciencia, Tecnología e Innovación y la Fundación Sadosky

M. P. Bonomini et al.

pecíficamente cuando la distancia es un obstáculo para los servicios de la salud” [1]. Según dichas definiciones, la telesalud comprende cuatro dimensiones: la telemedicina, la teleeducación, la teleinvestigación y la telegestión.

La telemedicina involucra a su vez un espectro de opciones para la atención y seguimiento médico mediado por las tecnologías informáticas. Dentro de éstas, se tiene la teleconsulta, o la búsqueda de información médica o asesoramiento por personal médico, la cual puede desarrollarse entre pacientes y profesionales de la salud o entre estos últimos. En la actualidad, la teleconsulta es la de mayor uso por ser la consulta médica la base de la práctica clínica de la medicina. La teleconsulta puede realizarse asincrónicamente, es decir, cuando la recopilación de información y su análisis por parte del profesional se realizan en momentos distintos, y sincrónica, en donde la consulta se realiza en tiempo real (por ejemplo mediante videoconferencia). Por otra parte, se tiene el *telemonitoreo*, que permite obtener información de rutina o especial con respecto a la condición del paciente -variables fisiológicas, imágenes, etc.- con el fin de monitoreo o seguimiento. Intuitivamente, el telemonitoreo ofrece oportunidades para identificar la escalada de síntomas y facilitar cuidados paliativos proactivos oportunos.

El telemonitoreo, como tal, es particularmente beneficioso para el cuidado en el hogar de pacientes con enfermedades crónicas [10, 11]. Por ejemplo, se estima que un 10% de la población mayor a 70 años padece insuficiencia cardíaca, mientras que un tercio de la población adulta padece hipertensión arterial. Así, las *plataformas de monitoreo* implementan a través de sistemas de software mecanismos que monitorean a distancia la presencia de síntomas o alteraciones en los signos vitales, lo que permite detectar las descompensaciones en etapas incipientes y facilitar su tratamiento evitando la saturación de los sistemas de salud. Esto permite reducir significativamente la mortalidad tanto como la tasa de reinternaciones y la adherencia terapéutica de los pacientes.

La adopción de sistemas de monitoreo, sin embargo, no está libre de desafíos [2], entre los que pueden mencionarse la usabilidad, ergonomía y facilidad de operación -sobre todo pensando en adultos mayores-, la redituabilidad económica, la durabilidad y robustez, y la compatibilidad con estándares establecidos tales como HL7 y FHIR [3]. Así, desde Virtual Sense S.A. y en conjunto con investigadores CONICET del IAM (Instituto Argentino de Matemática) e ISISTAN (Instituto Superior de Ingeniería de Software Tandil) se está desarrollando la primer y única plataforma de telemonitoreo que integra IoT (Internet of Things) y aprendizaje de máquina (IA) del país, con foco en varios de estos aspectos, y financiado por un subsidio de la Fundación Sadosky y el Ministerio de Ciencia, Tecnología e Innovación.

El propósito de la plataforma es permitir a los pacientes con diferentes patologías crónicas -actualmente la plataforma se enfoca en la insuficiencia cardíaca- recibir una atención adecuada desde el hogar. En su versión actual, el proyecto comprende tres objetivos técnicos, vale decir el desarrollo de: a) una aplicación móvil que utiliza el protocolo Bluetooth Low Energy para conectar a diferentes dispositivos de medición, incluido un ECG inalámbrico, b) el desarrollo de algoritmos de procesamiento de señales biomédicas como pueden ser los de detección

Plataforma IoT de telemonitoreo cardíaco

de fibrilación auricular o variabilidad de la frecuencia cardíaca para estimar frecuencia cardíaca o estrés en el paciente, y c) el desarrollo de un sistema de recomendación que permita automatizar la dosificación de los medicamentos basado en un sistema complejo de reglas aportados por los expertos en cardiología en este caso.

En este artículo, se reportan los resultados preliminares de este proyecto, con foco en describir el diseño, la evaluación y la validación de los algoritmos de detección de insuficiencias cardíacas implementados al momento. Así, la siguiente sección describe los algoritmos implementados y su rendimiento. Por su parte, la sección 3 brinda detalles de la arquitectura de software de la plataforma y las tecnologías involucradas en su estado actual. Finalmente, la sección 4 concluye el artículo y describe los pasos próximos del proyecto.

2 Detección de FA

Las plataformas de telemonitoreo, y en particular, aquellas que detectan patologías cardíacas, favorecen la toma de datos a los fines de análisis mediante smartphones o dispositivos portátiles. Esto resulta de gran utilidad ya que posibilita el monitoreo continuo en potencialmente un gran número de usuarios hogareños. Sin embargo, las señales de electrocardiograma (ECG) obtenidas mediante tales dispositivos suelen ser de baja calidad debido a la alta proporción de ruido. Esto se debe a que los dispositivos son portátiles, pequeños y por ende no tan sofisticados, y a que en la mayoría de los casos la señal no es tomada por un profesional médico sino por el propio paciente.

En el contexto del proyecto, se trabajó entonces en algoritmos para procesar estas señales como paso previo a realizar detección temprana de patologías cardiovasculares. Para poder procesar estas señales, es necesario primero detectar su calidad, permitiendo encontrar segmentos no ruidosos. De esta forma, la detección de calidad de señales cobra importancia y se vuelve clave para cualquier tipo de procesamiento posterior. Los segmentos no ruidosos son los que luego se procesan para realizar detección automática de enfermedades de forma exitosa, tales como la fibrilación auricular (FA). Para acotar el alcance, se hizo foco en la señal de electrocardiograma portátil de 1 derivación.

2.1 Base de datos

Los algoritmos aquí desarrollados se construyeron utilizando la base de datos Short Single-Lead AF Database (SSLAFDB) de Physionet [13]. Esta cuenta con 8528 señales de ECG de 1 derivación de duración entre 9 y 60 segundos, obtenidas mediante un dispositivo para smartphone. La frecuencia de muestreo de las señales es de 300 Hz. Estas señales se encuentran clasificadas en 4 clases según el ritmo predominante: N: Ritmo normal/sinusal, A: Fibrilación auricular, O: Otros ritmos y ~: Ruidosas.

M. P. Bonomini et al.

2.2 Desarrollo de algoritmo de detección de calidad de señales de ECG

El algoritmo de detección de calidad se basó en la incorporación de varios índices de ruido a un clasificador con salida binaria (segmento ruidoso/segmento no ruidoso). Estos índices se aplicaron sobre ventanas de la señal de 5 segundos de duración. Las ventanas fueron filtradas con un filtro de mediana para eliminar el ruido proveniente de la deriva. Luego, las ventanas fueron rectificadas y normalizadas.

El clasificador se entrenó con un conjunto de ventanas ruidosas y no ruidosas seleccionadas manualmente, con 400 ventanas representativas de las dos clases, resultando 800 en total. Las ventanas ruidosas fueron elegidas de señales ya clasificadas como ruidosas en SSLAFDB. Las ventanas no ruidosas se eligieron de señales clasificadas como N, A y O, para tener en cuenta todos los ritmos posibles. En la Figura 1 se muestran los gráficos de 4 ventanas como ejemplo de aquellas seleccionadas manualmente.

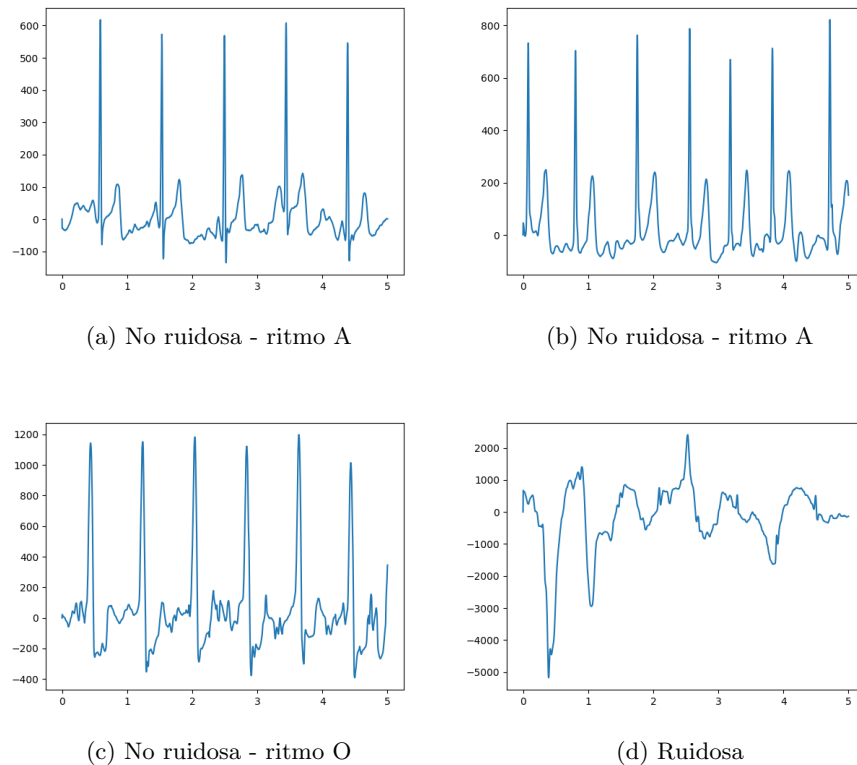


Fig. 1. Ejemplos de ventanas seleccionadas manualmente

Plataforma IoT de telemonitoreo cardíaco

El 20% de los datos se utilizaron para generar el conjunto de testeo y 80% para entrenamiento, realizando antes un mezclado aleatorio de los datos. Los modelos clasificadores considerados fueron K-nearest neighbors (KNN), support-vector machine (SVM) y regresión logística (RL). La efectividad de estos clasificadores se evaluó usando la métrica de exactitud, obtenida del promedio de un esquema de validación cruzada con $k=5$.

En la etapa de optimización de características se obtuvo el conjunto de índices más informativos para la clasificación:

- bSQI [5], el cual compara la detección realizada por dos detectores de complejos QRS, siendo uno mucho más sensible al ruido que el otro
- pSQI [6], que compara la energía presente en las bandas de frecuencia características de ECG y EMG
- kSQI [7], que se basa en la Kurtosis de la señal
- sSQI, que se basa en el skewness de la señal [14]

Utilizando estos índices, el clasificador más efectivo fue el RL, con una exactitud del 86.33%. En la Figura 2 se muestra la matriz de confusión obtenida para el clasificador RL.

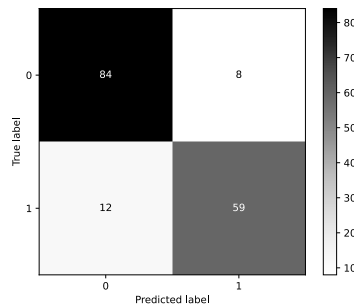


Fig. 2. Matriz de confusión para el conjunto de test generada mediante el clasificador RL. La etiqueta "0" equivale a ventana no ruidosa, y la etiqueta "1" a ruidosa

Posterior al entrenamiento, el algoritmo desarrollado realiza las predicciones, efectuando un suavizado por continuidad de las mismas para obtener las predicciones finales. Reunida esta información, se obtiene la clasificación de ruido de la señal segundo a segundo. Finalmente, se eliminan de la señal los segmentos detectados como ruidosos, y la señal restante se utiliza como entrada al detector detallado en la siguiente subsección.

M. P. Bonomini et al.

2.3 Desarrollo de algoritmo de detección de FA

Una vez más, el desafío en este punto es lograr una detección exacta y robusta con señales monoderivación y de menor calidad que los ECGs de reposo multi-derivación que se utilizan en la consulta médica. Para esto, se implementaron múltiples índices de FA que alimentaron un clasificador para reconocer la presencia o ausencia de FA. Cabe destacar que la clasificación final del sistema es binaria, por lo que los ritmos N y O fueron fusionados en una única clase 0, y los A formaron la clase 1. A continuación se detalla la metodología utilizada en esta sección.

Inicialmente, se aplicó un algoritmo de detección de complejos QRS a los segmentos, basado en el algoritmo de Pan y Tompkins [9]. En la Figura 3 se muestran los complejos QRS (rojo) detectados en una señal ejemplo. A partir de la detección de complejos QRs, se calcularon las series RR [8] que evidencian la estructura de la señal en base al tiempo transcurrido entre dos latidos consecutivos. Luego, La serie RR de cada segmento se obtuvo mediante la distancia temporal de los complejos QRS. Las series RR de los segmentos no ruidosos de cada señal se concatenaron para obtener la serie RR.

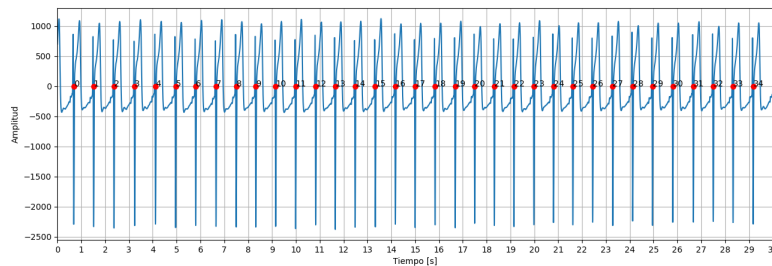


Fig. 3. Ejemplo de detección de complejos QRS en una señal de ECG

Sobre las series RR se calcularon un conjunto de índices de FA basados en diferentes análisis, como histogramas, entropía, puntos de inflexión, gráficos de Poincaré o coherencia, entre otros. Los índices implementados se utilizaron como atributos para entrenar un modelo de aprendizaje supervisado para detección de FA. Para esto, se aplicaron los modelos SVM y RL, que fueron introducidos en la sección anterior.

La etapa de optimización de características del modelo consistió en elegir la mejor combinación de los índices arriba mencionados en cuanto a poder de clasificación. En esta etapa se realizó un análisis estadístico de ANOVA factorial y chi cuadrado, evaluando el rendimiento del clasificador entrenado con distintas combinaciones de índices. Como resultado se eligió una combinación de 5 índices, los cuales se detallan a continuación:

Plataforma IoT de telemonitoreo cardíaco

- P_{NMASD} (Normalized mean of absolute successive differences): La media normalizada de las diferencias absolutas sucesivas de la serie RR; un parámetro basado en dispersión estadística.
- $Diff_{50}$: El porcentaje de intervalos RR cuyo intervalo RR siguiente difiere en más de 50 ms; otro parámetro basado en dispersión estadística.
- $Poin_2$: Un parámetro obtenido a partir del gráfico de Poincaré $(\Delta RR(n), \Delta RR(n+1))$.
- $Poin_3$: Un parámetro obtenido a partir del gráfico de Poincaré $(RR(n), \Delta RR(n))$.
- I_{SampEn} (Coefficient of sample entropy): El coeficiente de la entropía de muestreo es una medida de entropía relacionada con la probabilidad de que una señal se repita a sí misma.

Gráficos de Poincaré

Los gráficos de Poincaré son gráficos de dispersión de pares sucesivos de intervalos RR, utilizados para la caracterización de distintos tipos de ritmos cardíacos. En particular, el índice seleccionado $Poin_2$ se obtiene a partir de una variante del gráfico de Poincaré definido por $(\Delta RR[n], \Delta RR[n+1])$, donde $\Delta RR[n] = RR[n+1] - RR[n]$. El índice $Poin_2$ es un número equivalente al número total de los bins de las regiones del gráfico que contienen al menos 1 (un) dato, excluyendo la región del origen que tiende a estar poblada por puntos pertenecientes al ritmo normal. Entonces, un valor alto de $Poin_2$ sugiere la presencia de FA debido al alto grado de variabilidad de la serie RR. Por otro lado, el índice $Poin_3$ proviene de un gráfico de Poincaré de tipo $(RR[n], \Delta RR[n])$. En este caso, el índice $Poin_3$ se calcula simplemente contando el número de bins de todas las regiones del gráfico que contienen al menos 1 dato. Al igual que para $Poin_2$, un alto valor de $Poin_3$ es indicador de FA.

Coeficiente de la entropía de muestreo

El coeficiente de la entropía de muestreo se basa en la entropía de muestreo según la siguiente fórmula:

$$I_{CSampEn} = I_{SampEn} + \ln(2r) - \ln(\bar{m}_x) \quad (1)$$

Donde r es una tolerancia fijada experimentalmente en 0.05, $\bar{m}_x$ es la media de la serie RR, y I_{SampEn} es la entropía de muestreo. La entropía de muestreo se define como el logaritmo natural de la probabilidad condicional de que una señal se repita a sí misma por $m+1$ muestras dentro de una tolerancia r , dado que se repite a sí misma por m muestras. El valor utilizado de m fue de 1 ya que resultaba en una mejor separación de clases. Esta entropía se calcula de la siguiente forma:

$$I_{SampEn} = -\ln \left(\frac{B(m+1, r)}{B(m, r)} \right) \quad (2)$$

Donde $B(m, r)$ es la probabilidad de que pares de secuencias coincidan por m muestras. Un valor pequeño de I_{SampEn} sugiere que la señal se repite a sí misma y es por ende regular.

M. P. Bonomini et al.

En la siguiente tabla pueden verse los resultados obtenidos para la validación utilizando estos 5 índices. En la primer columna se utilizaron solo ritmos A y N para entrenar y testear. En la columna de la derecha se muestran los resultados agregando ritmos O al conjunto de datos y clasificando de forma dicotómica entre FA y no FA. Todas las métricas reportadas fueron obtenidas realizando validación cruzada con 5 particiones.

	Ritmos N y A	Ritmos N, A y O
Exactitud	92.74	86.62
Precisión	94.90	88.35
Especificidad	95.11	88.84
Sensibilidad	90.37	84.4

2.4 Validación externa

El equipo informático de VirtualSense S.A., ajeno al equipo de desarrollo del algoritmo de detección de FA, realizó una validación del modelo resultante con bases de datos externas a la de entrenamiento.

Las bases de datos utilizadas para la validación fueron:

- Kaggle [16]: Esta base de datos posee 6528 registros, dentro de los cuales 1514 son de fibrilación auricular, 2000 de ritmo sinusal y 3014 de otros ritmos. Todas las señales tienen 10 segundos de duración y fueron obtenidas con una frecuencia de muestreo de 500 Hz.
- Base de datos sintética [17]: Set de 10.000 registros de ECG balanceados, 5000 ritmo sinusal y 5000 FA. Sintetizados a partir de un modelo para simular ECG con arritmias a una frecuencia de muestreo de 250 Hz.

En la siguiente tabla se muestran las métricas de exactitud, precisión, especificidad y sensibilidad, de la validación externa realizada por VirtualSense S.A. con la base de datos de Kaggle y la sintética.

	Kaggle	DB sintética
Exactitud	93.14	89.53
Precisión	94.08	84.09
Especificidad	95.78	81.54
Sensibilidad	89.61	97.52

2.5 Actualizaciones del modelo

A lo largo del desarrollo se realizaron varias actualizaciones que mejoraron el poder de clasificación correcta del modelo. Algunas de estas mejoras se detallan a continuación.

Plataforma IoT de telemonitoreo cardíaco

- Agregado de ritmos. En un inicio solo se operó con señales de ritmo N y A, pero luego se agregaron señales de ritmo O. Estas se fusionaron con las señales ritmo N, formando la clase de ausencia de FA. Como era de esperarse, al agregar estos ritmos el rendimiento del clasificador empeoró. Esto se debe a que los ritmos O son un conjunto de ritmos que incluyen otros tipos de arritmias. Al ser variadas, estas tienen características distintas, a veces más similares a las ritmo A y otras veces a las ritmo N, lo que confunde al clasificador. De todas formas, usando un conjunto de 5 índices de FA, se logró caracterizar cada clase, haciendo que la exactitud del clasificador ronde el 90 %.
- Balance de clases. Como se comentó anteriormente, la proporción de señales de ritmo A en la base de datos SSLAFDB no llega al 10 %. Esto hace que si el modelo se entrena con un 80 % de los datos de la base de datos completa, el modelo quede sesgado a detectar mayor cantidad de señales con ausencia de FA. En una de las actualizaciones del modelo, esto fue corregido igualando el número de señales sin FA al de las de ritmo A en el conjunto de entrenamiento.
- Acondicionamiento de las señales. Previamente las señales se filtraban únicamente con un filtro pasabandas de frecuencias de corte de 8 y 25 Hz para reducir el ruido existente. Para reducir aún más la deriva de la señal, se agregaron los filtros de mediana. Adicionalmente, se añadió una etapa en la que se corrige la polaridad de la señal. Esto se realiza generando un template mediana del complejo QRS de cada señal, y comparando sus puntos máximos y mínimos. Finalmente, se agregó una normalización de la amplitud de la señal, para que todas las señales tengan rangos de amplitud similar, entre -1 y 1, aproximadamente.
- Clasificador de ruido. Para clasificar si cada ventana de 1 segundo es ruidosa se recorre la señal con una ventana móvil de 5 segundos con corrimientos de 1 segundo con superposición. Inicialmente, el resultado de si cada ventana era ruidosa o no, se daba de forma binaria (0 o 1), y luego se realizaba el promedio ponderado descrito anteriormente para calcular si cada segundo era ruidoso. Para obtener un valor final más preciso, se cambió el resultado de las ventanas de 5 segundos, de la clasificación binaria a un valor entre 0 y 1 que representa la probabilidad de una ventana de ser ruidosa. Después de calcular el promedio ponderado para cada segundo, se realiza un redondeo de esta probabilidad, obteniéndose la clasificación final de ruido en valores de 0 y 1 segundo a segundo. Por este mismo motivo, el clasificador utilizado fue cambiado de SVM a RL.

3 Arquitectura y diseño de la plataforma

Se ha desarrollado una primer versión de la plataforma, que incluye una aplicación móvil desarrollada en Flutter y un backend en PHP y Laravel. Flutter [12] es un framework open source de Google que permite construir aplicaciones móviles multiplataforma utilizando el lenguaje Dart, y compilar el código

M. P. Bonomini et al.

de forma nativa a diferentes plataformas móviles (Android, iOS) y Web. Actualmente, se ha producido una versión para Android⁸.

El backend, por su parte, expone a través de una API Rest diversos servicios para recepcionar mediciones realizadas por los dispositivos y comunicadas por la app. Estas mediciones se almacenan en el backend y se asocian a un paciente, el cual se identifica de forma unívoca a través de las credenciales de acceso a la app. Además, en el backend existen servicios que, mediante Docker, ejecutan los algoritmos de detección de ruido en señal ECG y fibrilación auricular, los cuales están implementados en Python.

En lo que respecta a la app en sí, la misma integra diferentes SDK que implementan los mecanismos específicos para comunicarse, con los dispositivos de medición soportados. Al momento, esto incluye:

- Una balanza (<https://play.google.com/store/apps/details?id=com.chipsea-btcontrol.en>) de la empresa OKOK International.
- Un tensiómetro (<https://www.linkedin.com/company/joytech-healthcare>) de la empresa JoyTech Health Care Inc.
- Un pulsioxímetro de la empresa Yonkercare (<https://www.yonkercare.com/>).
- Un ECG de la empresa Borsam Biomedical Instruments (<http://en.wecardio.com/>).

En algunos casos, se integró un SDK existente provisto por la empresa, y en otros, se implementó como parte del proyecto el SDK a partir de la especificación de bajo nivel del protocolo BLE específico del dispositivo. Cabe destacar que los SDK están implementados en código nativo, por lo que la comunicación Dart-SDK se realiza utilizando el mecanismo de Method Channel [12]. Este mecanismo permite llamar desde Flutter a funciones nativas, por ejemplo, código Java o Kotlin para SDK versión Android o código Swift or Objective-C para SDK versión iOS.

A nivel implementación, la app reusa bibliotecas públicas, todas disponibles para Android e iOS, para incluir un mecanismo de reintentos de invocación a las APIs Rest, soporte para persistencia de llamados fallidos, y reintento diferido de éstos últimos:

- **dio** (<https://pub.dev/packages/dio>): Esta biblioteca representa un wrapper al soporte básico HTTP de Flutter, y extiende el mismo con facilidades para comunicación Rest. En particular, se aprovechó el soporte de *interceptores*, y puntualmente se utiliza el *RetryInterceptor*, que permite por cada invocación a la API realizar reintentos en caso de falla (ya sea por falta de conectividad de la aplicación, API no responsiva, error interno de la API, etc.).
- **get_storage** (https://pub.dev/packages/get_storage): Es una biblioteca rápida y liviana que administra una base de datos clave-valor en memoria, y realiza backups de los datos a disco. Se utiliza para grabar cada invocación previo realizar la misma, y en caso de éxito, se elimina de la base de datos. Si una invocación falla, de acuerdo al protocolo de reintentos anterior, quedará

⁸ <https://play.google.com/store/apps/details?id=com.virtualsense.pacientes>

Plataforma IoT de telemonitoreo cardíaco

la misma persistida en la base de datos, registrando por lo tanto el dato no enviado, y la fecha y hora original cuando se produjo el dato.

- **neat_periodic_task** (https://pub.dev/packages/neat_periodic_task): Esta biblioteca permite a nivel Flutter la ejecución periódica de tareas genéricas. Se utiliza para lanzar un scheduler y chequear periódicamente por invocaciones a la API ejemplo que resultaron no exitosos, y en consecuencia reintentar su envío. Para esto, se consulta el contenido de la base de datos anterior, borrando las entradas que pueden ser enviadas exitosamente.

4 Conclusiones

En este artículo se han descrito los avances hacia una plataforma de software para telemonitoreo, que apunta a permitir un monitoreo permanente de pacientes, lo que posibilita detectar emergencias o situaciones de riesgo, reducir los tiempos de traslado y riesgos de contagios intrahospitalarios. Asimismo, la plataforma permitirá a las instituciones médicas atender a un número mayor de pacientes, a un menor costo, y con mayor eficiencia en los resultados. La plataforma constituye un aporte para mejorar el acceso al sistema de salud de la población en principio de Argentina que padece patologías crónicas, debido a que permite el acceso a los profesionales más capacitados de acuerdo a cada patología, independientemente de su localización territorial.

En una primera etapa, la plataforma buscará brindar el servicio a pacientes que sufren insuficiencia cardíaca y han tenido una internación en los últimos 12 meses, ubicados en el Área Metropolitana de Buenos Aires. Como pasos subsiguientes, se validarán los resultados de los algoritmos mediante señales de ECG reales, en base al dispositivo ECG ya integrado (ECG WeCardio de la empresa Borsam). Con estos datos de pacientes reales, se podrán además analizar ciertas métricas de performance, como sensibilidad y especificidad versus los distintos niveles de ruido presentes en la señal. Además, se ampliará la plataforma para atender pacientes con hipertensión arterial, diabetes, parkinson y otras enfermedades crónicas, para lo que se profundizará en la integración con dispositivos de medición necesarios, se expandirá la funcionalidad de la app, y se desarrollará la algoritmia necesaria. Por otra parte, debido a que el backend ya cuenta con soporte de almacenamiento de datos, se está diseñando una interfaz bajo los estándares HL7 y FHIR (<https://hl7.org.ar/site/home>) para intermediar entre clientes y backends externos (como pueden ser sistemas de hospitales e historia clínica) para permitir acceder a dichos datos de acuerdo a los principios de interoperabilidad de dichos estándares. En principio, esta interfaz solo aplicará a los datos observados de los dispositivos de medición y los datos de pacientes, para luego expandirse a todo el espectro de los datos que almacena la plataforma.

5 Agradecimientos

Agradecemos a la Lic. Mariela Ruffino y al Ing. Leandro Batlle de la Fundación Sadosky por el soporte de gestión brindado al proyecto, y a la Fundación Sadosky y al MinCyT por el fondo otorgado.

M. P. Bonomini et al.

References

1. Ministerio de Salud de la República Argentina (2023), "Preguntas frecuentes sobre Telesalud". <https://www.argentina.gob.ar/salud/telesalud> (último acceso Abril de 2023).
2. Hashemi, A., Nourbakhsh, S., Tehrani, P., and Karimi, A. (2018), "Remote tele-monitoring of cardiovascular patients: Benefits, barriers, new suggestions". *Artery Research*, 22, 57-63.
3. Setyawan, R., Hidayanto, A. N., Sensuse, D. I., Suryono, R. R., and Abilowo, K. (2021), "Data Integration and Interoperability Problems of HL7 FHIR Implementation and Potential Solutions: A Systematic Literature Review". 5th IEEE International Conference on Informatics and Computational Sciences (ICICoS) (pp. 293-298).
4. Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Tibshirani, R., and Friedman, J. (2009), Overview of supervised learning. *The elements of statistical learning: Data mining, inference, and prediction*, 9-41.
5. Liu, F., Liu, C., Zhao, L., Jiang, X., Zhang, Z., Li, J., Wei, S. and Zhang, Y. (2018), "Dynamic ECG signal quality evaluation based on the generalized bSQI index". *IEEE Access*, 6, 41892-41902.
6. Qin, Q., Li, J., Yao, S., Liu, C., Huang, H., and Zhu, Y. (2019), "Electrocardiogram of a silver nanowire based dry electrode: quantitative comparison with the standard Ag/AgCl gel electrode". *IEEE Access*, 7, 20789-20800.
7. Rahman, S., Karmakar, C., Natgunanathan, I., Yearwood, J., and Palaniswami, M. (2022), "Robustness of electrocardiogram signal quality indices". *Journal of the Royal Society Interface*, 19(189), 20220012.
8. Singh, D., Vinod, K., and Saxena, S. C. (2004), "Sampling frequency of the RR interval time series for spectral analysis of heart rate variability". *Journal of medical engineering & technology*, 28(6), 263-272.
9. Fariha, M., Ikeura, R., Hayakawa, S., and Tsutsumi, S. (2020), "Analysis of Pan-Tompkins algorithm performance with noisy ECG signals". *Journal of Physics: Conference Series* (Vol. 1532, No. 1, p. 012022). IOP Publishing.
10. Merchant, M. H., Alsalem, A. B., Warnock, M., Wirth, D., and Ogunniyi, M. O. (2018), "The Impact of Telemonitoring on 30-day Readmissions for Patients Hospitalized with Heart Failure: A Safety-Net Hospital's Experience". *Journal of Cardiac Failure*, 24(8), S93.
11. Koehler, F., Koehler, K., Deckwart, O., Prescher, S., Wegscheider, K., Kirwan, B., Winkler, S., Vettorazzi, E., Bruch, L., Oeff, M., and others (2018), "Efficacy of telemedical interventional management in patients with heart failure (TIM-HF2): a randomised, controlled, parallel-group, unmasked trial". *The Lancet*, 392(10152), 1047-1057.
12. Windmill, E. (2020), "Flutter in action". Simon and Schuster.
13. PhysioNet (2017), "AF Classification from a short single lead ECG recording: the PhysioNet/Computing in Cardiology Challenge 2017". <https://archive.physionet.org/challenge/2017/> (último acceso Abril de 2023).
14. Zhao, Z. and Zhang, Y. (2018), "SQI Quality Evaluation Mechanism of Single-Lead ECG Signal Based on Simple Heuristic Fusion and Fuzzy Comprehensive Evaluation". *Frontiers in Physiology*, vol. 9, no. 727.
15. Leif, S. (2018), "Atrial fibrillation from an engineering perspective". Cham, Switzerland: Springer.

Plataforma IoT de telemonitoreo cardíaco

16. Kaggle Inc. (2023), "PTB-XL - Atrial Fibrillation Detection". <https://www.kaggle.com/datasets/arjunascagnetto/ptb-xl-atrial-fibrillation-detection?resource=download> (último acceso Abril de 2023)
17. Sološenko, A., Petrėnas, A., Paliakaitė, B., Marozas, V., & Sörnmo, L. (2022). Model for Simulating ECG and PPG Signals with Arrhythmia Episodes (version 1.3.1). PhysioNet.